

# Fondamenti di Informatica C

## *Esercitazioni di Laboratorio / 4*

<http://polaris.ing.unimo.it/fiC/laboratorio.html>

---

*Ing. Francesco De Mola*

[demola.francesco@unimore.it](mailto:demola.francesco@unimore.it)

→ Ricevimento su appuntamento ←

DII, Modena – Via Vignolese  
(lab. Dottorandi 1° piano)  
Tel.: 059 205.61.42

DISMI, Reggio Emilia – Via  
Amendola 2 (Pad. Morselli)  
Tel.: 0522 52.26.60

Home Page:

<http://www.agentgroup.ing.unimo.it/didattica/curriculum/francesco>

## Outline

- Sorgenti e formati, a byte e caratteri:
  - o Ripasso
  - o Esempi
- Gestione eccezioni: `try...catch`
- Esercizio1: lettura/scrittura file binario
- Esercizio2: gestione famiglie su file

# Sorgenti e formati

Tipologia stream input/output: byte o caratteri?

→ Classi astratte:

InputStream/OutputStream e  
Reader/Writer

Sorgente: dove leggere/scrivere?

→ Classi sorgente

Fondamenti di Informatica C - Esercitazione 4

3

Formati: come leggere/scrivere?

## Stream di byte

	<i>InputStream</i>	<i>OutputStream</i>
<b>DOVE</b> (classi sorgenti)		
→ File	FileInputStream	FileOutputStream
→ Array di byte	ByteArrayInputStream	ByteArrayOutputStream
<b>COSA</b> (classi filtraggio)		
→ Dati	DataInputStream	DataOutputStream
→ Oggetti *	ObjectInputStream	ObjectOutputStream
* Le classi da salvare su file devono implementare l'interfaccia <i>Serializable</i>		
→ Tramite buffer	BufferedInputStream	BufferedOutputStream

Fondamenti di Informatica C - Esercitazione 4

4

# Stream di caratteri

	<i>Reader</i>	<i>Writer</i>
<b>DOVE</b> ( <i>classi sorgenti</i> )		
→ Tastiera/Monitor	InputStreamReader	OutputStreamWriter
→ File	FileReader	FileWriter
→ Array di caratteri	CharArrayReader	CharArrayWriter
<b>COSA</b> ( <i>classi di filtraggio</i> )		
→ Tramite buffer	BufferedReader	BufferedWriter
→ Dati? ( <i>astratte:</i> )	<i>FilterReader</i>	<i>FilterWriter</i>

Fondamenti di Informatica C - Esercitazione 4

5

## Esempi

- Lettura di dati formattati da file binario:

```
FileInputStream f_in = new
    FileInputStream("nomeFile.dat");
DataInputStream in = new DataInputStream(f_in);
```

=

```
DataInputStream in = new DataInputStream(
    new FileInputStream("nomeFile.dat") );
```

- Scrittura di byte bufferizzati su file testuale:

```
BufferedWriter out = new BufferedWriter(
    new FileWriter("nomeFile.txt"));
```

- Lettura di caratteri bufferizzati da tastiera:

```
BufferedReader in = new BufferedReader(
    new InputStreamReader(System.in) );
```

Fondamenti di Informatica C - Esercitazione 4

6

# Gestione eccezioni

Le istruzioni che possono generare errori, in Java lanciano eccezioni, oggetti derivati dalla classe `Exception`.

Devono essere opportunamente gestite:

```
BufferedReader in;  
...  
try{  
    String line = in.readLine();  
  
}catch(IOException e){  
  
    System.out.println("!Errore in lettura");  
    e.printStackTrace();  
}
```

Fondamenti di Informatica C - Esercitazione 4

7

## Esercizio1: **Scrittura di tipi di dato primitivi, letti da stdin, su file binario**

Realizzare un menù che contempli 3 scelte:

- 1) Scrivi
- 2) Leggi
- 3) Esci

L'opzione '**Scrivi**' chiede all'utente di inserire un *intero* e un *double*, e li salva su file binario.

L'opzione '**Leggi**' estrae i dati dal file binario e li visualizza a video.

L'opzione '**Esci**' termina il programma non senza aver salutato l'utente.

Fondamenti di Informatica C - Esercitazione 4

8

**Esercizio2: Gestione di famiglie e persone su file binario e testuale**

Si creino le classi necessarie alla rappresentazione del concetto di "*famiglia*", in particolar modo modellando ogni *persona* come un oggetto a se stante: prevedere gli attributi necessari alla memorizzazione dell'*indirizzo* della famiglia e dei *componenti*, mentre per ogni persona informazioni come il *nome*, il *cognome*, il  *Sesso*, ecc.

Sia la classe Famiglia che Persona definiscono dei metodi per la stampa su stdout del loro stato.

Si metta a disposizione una **interfaccia generica per la scrittura/lettura** di ogni famiglia su un file esterno, sia di testo che binario.

Si forniscano implementazioni concrete per la **scrittura/lettura binaria e testuale**.